Introduction

Pneumonia is a lung infection that can be caused by bacteria, viruses, or fungi. It is a leading cause of death worldwide, especially among children and the elderly. Early diagnosis and treatment are essential for reducing the risk of complications and death. Chest X-rays are a common diagnostic tool for pneumonia. However, interpreting chest X-rays can be challenging, even for experienced radiologists. This is because the signs of pneumonia on a chest X-ray can be subtle and vary depending on the type of pneumonia.

Convolutional neural networks (CNNs) are a type of deep learning algorithm that has been shown to be adequate for image classification tasks. In recent years, CNNs have been used to develop automated systems for the diagnosis of pneumonia from chest X-rays.

This project aims to develop a web application by a CNN model that can classify X-Ray images as a pneumonia case or a normal case. The model will be trained on a dataset of chest X-ray images that have been labeled as either pneumonia or normal. The model will be evaluated on a separate dataset of chest X-ray images. The successful development of this model could significantly impact the diagnosis and treatment of pneumonia. The model could be used to automate the diagnosis of pneumonia, which could free up radiologists to focus on more complex cases. Additionally, the model could be used to improve the accuracy of the diagnosis of pneumonia, which could lead to earlier treatment and a reduction in the risk of complications and death.

Literature Review

There has been a growing body of research on the use of CNNs for the diagnosis of pneumonia from chest X-rays. In a recent study, a CNN model was trained on a dataset of 10,000 chest X-ray images and was able to achieve an accuracy of 92% in classifying pneumonia cases.

Another study compared the performance of CNNs with the performance of radiologists in diagnosing pneumonia from chest X-rays. The study found that CNNs were able to achieve an accuracy that was comparable to that of radiologists. These studies suggest that CNNs have the potential to be a valuable tool for the diagnosis of pneumonia from chest X-rays. However, more research is needed to improve the accuracy of CNN models and validate their performance in clinical settings.

Methods

1. The dataset: Dataset

The dataset that we are going to use for the image classification is Chest X-Ray images, which consist of 2 categories, Pneumonia and Normal. This dataset was published by Paulo Breviglieri, a revised version of Paul Mooney's most popular dataset. This updated version of the dataset has a more balanced distribution of the images in the validation set and the testing set. The data set is organized into 3 folders (train, test, value) and contains subfolders for each image category Opacity(viz. Pneumonia) & Normal.

Total number of observations (images): 5,856
Training observations: 4,192 (1,082 normal cases, 3,110 lung opacity cases)
Validation observations: 1,040 (267 normal cases, 773 lung opacity cases)
Testing observations: 624 (234 normal cases, 390 lung opacity cases)

2. Data Preparation:

A) Data Augmentation
B) Loading the images

3. CNN

A) Begin with a lower filter value such as 32 and begin to increase it layer wise.
Construct the model with a layer of Conv2D followed by a layer of MaxPooling.
The kernel_size is preferred to be odd number like 3x3.
Tanh, relu, etc. can be used for activation function, but relu is the most preferred activation function.
input_shape takes in image width & height with last dimension as color channel.
Flattening the input after CNN layers and adding ANN layers.
Use activation function as softmax for the last layer If the problem is more than 2 classes, define units as the total number of classes and use sigmoid for binary classification and set unit to 1.

B) Building the CNN Model sure looks like:

```
cnn = Sequential()
cnn.add(Conv2D(32, (3, 3), activation="relu", input_shape=(img_width, img_height, 1)))
cnn.add(MaxPooling2D(pool_size = (2, 2)))
cnn.add(Conv2D(32, (3, 3), activation="relu", input_shape=(img_width, img_height, 1)))
cnn.add(MaxPooling2D(pool_size = (2, 2)))
cnn.add(Conv2D(32, (3, 3), activation="relu", input_shape=(img_width, img_height, 1)))
cnn.add(MaxPooling2D(pool_size = (2, 2)))
cnn.add(Conv2D(64, (3, 3), activation="relu", input_shape=(img_width, img_height, 1)))
cnn.add(MaxPooling2D(pool_size = (2, 2)))
cnn.add(Conv2D(64, (3, 3), activation="relu", input_shape=(img_width, img_height, 1)))
cnn.add(MaxPooling2D(pool_size = (2, 2)))
cnn.add(Flatten())
cnn.add(Dense(activation = 'relu', units = 128))
cnn.add(Dense(activation = 'relu', units = 64))
cnn.add(Dense(activation = 'sigmoid', units = 1))
cnn.compile(optimizer = 'adam', loss = 'binary_crossentropy', metrics = ['accuracy'])
```

C) Fitting the model
EarlyStopping is called to stop the epochs based on some metric(monitor) and conditions (mode, patience) . It helps to avoid overfitting the model. Over here we are telling to stop based on val_loss metric, we need it to be minimum. patience says that after a minimum val_loss is achieved then after that in next iterations if the val_loss increases in any the 3 iterations then the the training will stop at that epoch.

Reduce learning rate when a metric has stopped improving. Models often benefit from reducing the learning rate by a factor of 2–10 once learning stagnates. This callback monitors a quantity and if no improvement is seen for a 'patience' number of epochs, the learning rate is reduced. Source

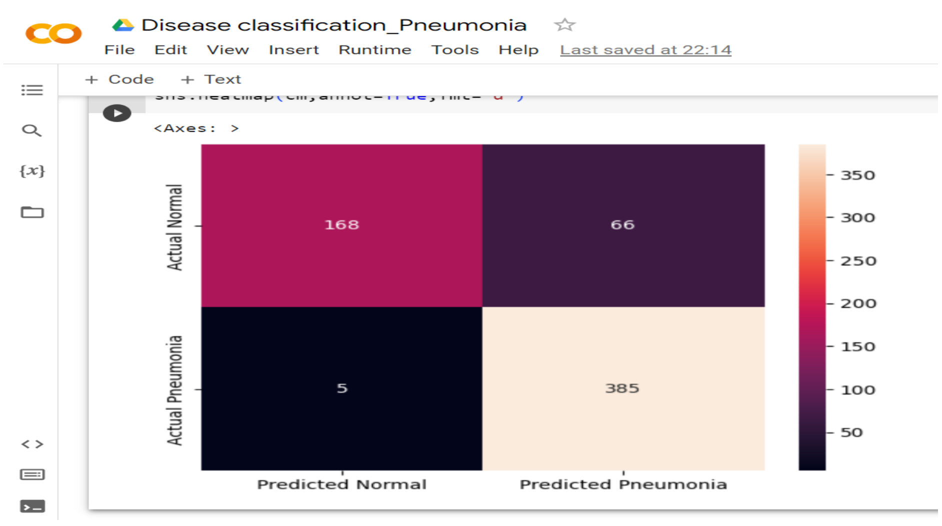
```
//Code
early = EarlyStopping(monitor=”val_loss”, mode=”min”, patience=3)
learning_rate_reduction = ReduceLROnPlateau(monitor=’val_loss’, patience = 2,
verbose=1,factor=0.3, min_lr=0.000001)
callbacks_list = [ early, learning_rate_reduction]
```

D) Model Training:
The parameters we are passing to model.fit are train set, epochs as 25, validation set used to calculate val_loss and val_accuracy, class weights and callback list.

```
//Code
cnn.fit(train,epochs=25, validation_data=valid, class_weight=cw,
callbacks=callbacks_list)
```

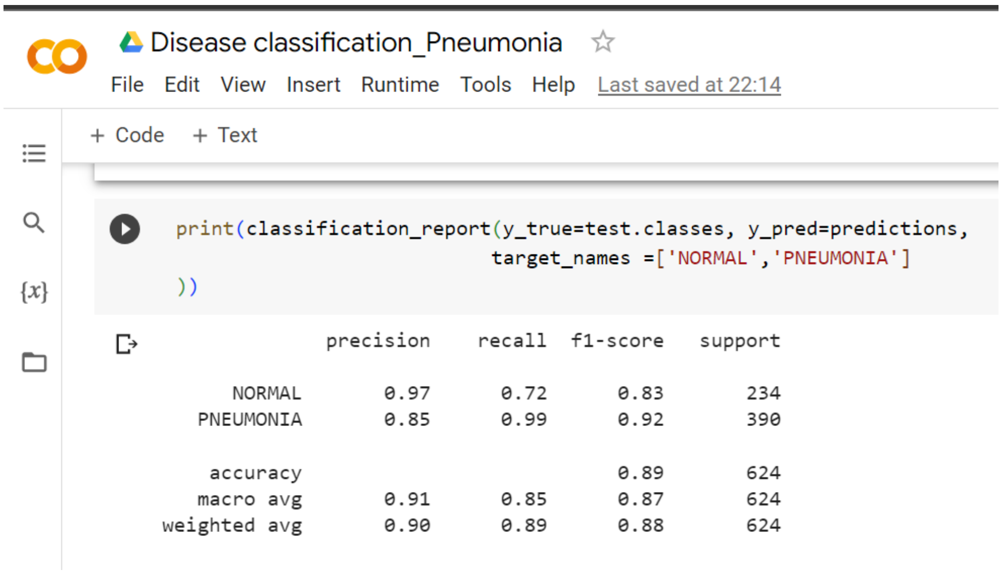
4. VISUALIZATION

E) Visualising with the help of confusion matrix:



```
<Axes: >
```

F) Classification report:


Disease classification_Pneumonia
File Edit View Insert Runtime Tools Help Last saved at 22:14
+ Code + Text

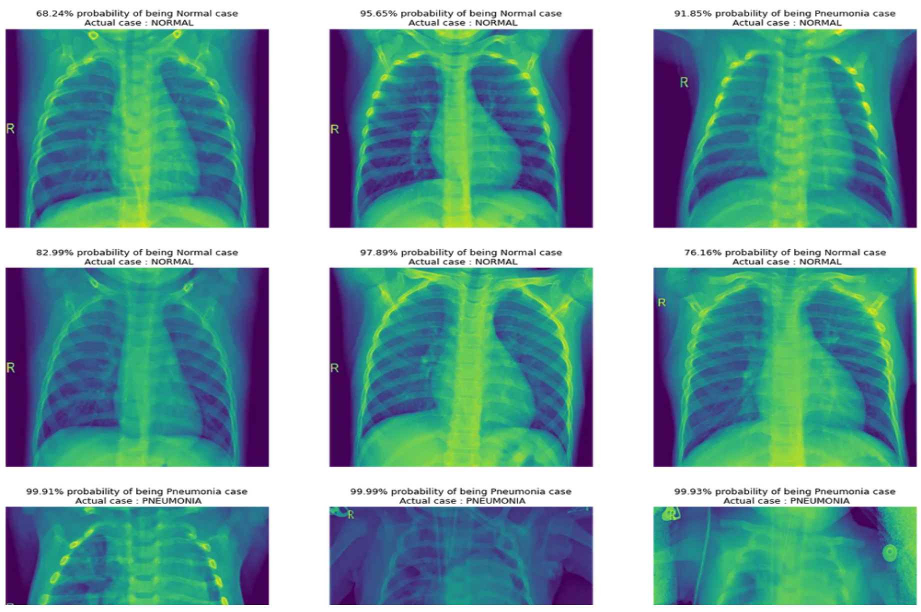
```
print(classification_report(y_true=test.classes, y_pred=predictions,
                            target_names =['NORMAL','PNEUMONIA']
))
```

```
              precision    recall  f1-score   support

      NORMAL       0.97      0.72      0.83       234
   PNEUMONIA       0.85      0.99      0.92       390

    accuracy                           0.89       624
   macro avg       0.91      0.85      0.87       624
weighted avg       0.90      0.89      0.88       624
```

G) Visualizing the predicted images with respective %:


68.24% probability of being Normal case
Actual case : NORMAL
R
95.65% probability of being Normal case
Actual case : NORMAL
R
91.85% probability of being Pneumonia case
Actual case : NORMAL
R
82.99% probability of being Normal case
Actual case : NORMAL
R
97.89% probability of being Normal case
Actual case : NORMAL
R
76.16% probability of being Normal case
Actual case : NORMAL
R
99.91% probability of being Pneumonia case
Actual case : PNEUMONIA
99.99% probability of being Pneumonia case
Actual case : PNEUMONIA
R
99.93% probability of being Pneumonia case
Actual case : PNEUMONIA

5. WEBAPP DEPLOYMENT

The web application was built and deployed by streamlit that can be found here: https://diseaseclassificationmadebysaswati.streamlit.app/

6. CONCLUSION

The development of a CNN model that can classify X-Ray images as a pneumonia case or a normal case has the potential to have a significant impact on the diagnosis and treatment of pneumonia. The model could be used to automate the diagnosis of pneumonia, which could free up radiologists to focus on more complex cases. Additionally, the model could be used to improve the accuracy of the diagnosis of pneumonia, which could lead to earlier treatment and a reduction in the risk of complications and death.

This project will contribute to the growing body of research on the use of CNNs for the diagnosis of pneumonia from chest X-rays. The results of this project will help to inform the development of future CNN models for the diagnosis of pneumonia.